



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ

«Информатика и системы управления»

КАФЕДРА

«Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 13

Студент: Керимов А. Ш.

Группа: ИУ7-64Б

Преподаватель: Толпинская Н. Б.

Москва.
2020 г.

Цель работы — получить навыки построения модели предметной области, разработки и оформления программы на Prolog, изучить принципы, логику формирования программы и отдельные шаги выполнения программы на Prolog.

Задание. Составить программу, т.е. модель предметной области – базу знаний, объединив в ней информацию – знания:

- **«Телефонный справочник»:** Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв),
- **«Автомобили»:** Фамилия_владельца, Марка, Цвет, Стоимость, и др.,
- **«Вкладчики банков»:** Фамилия, Банк, счет, сумма, др.

Владелец может иметь несколько телефонов, автомобилей, вкладов (Факты).

Используя правила, обеспечить возможность поиска:

1. **а)** По № телефона найти: Фамилию, Марку автомобиля, Стоимость автомобиля (может быть несколько),
в) Используя сформированное в пункте **а)** правило, по № телефона найти: только Марку автомобиля (автомобилей может быть несколько),
2. Используя простой, не составной вопрос: по Фамилии (уникальна в городе, но в разных городах есть однофамильцы) и Городу проживания найти: Улицу проживания, Банки, в которых есть вклады и №телефона.

Для задания1 и задания2:

для одного из вариантов ответов, и для **а)** и для **в)**, **описать словесно** порядок поиска ответа на вопрос, указав, как выбираются знания, и, при этом, **для каждого этапа унификации, выписать подстановку** – наибольший общий унификатор, и **соответствующие примеры** термов.

Практическая часть

Листинг 1. Телефонный справочник, автомобили и вклады

```
domains
    surname, phone, city, street, house, apartment = symbol
    address = address(city, street, house, apartment)
    model, color, cost = symbol
    bank, account, sum = symbol

predicates
    subscriber(surname, phone, address)
    car(surname, model, color, cost)
    deposit(surname, bank, account, sum)

    find_car_by_phone(phone, surname, model, cost)
    find_model_by_phone(phone, model)
    find_bank_street_by_surname_city(surname, city, bank, street)

clauses
    subscriber("Kerimov", "1111111111", address("Lyubertsy", "Yubileynaya", "17", "2")).
```

```

subscriber("Yugay", "222222222", address("Lyubertsy", "Yubileynaya", "17b", "18")).
subscriber("Ivanov", "333333333", address("Moscow", "Maroseyka", "12", "5")).
subscriber("Novikov", "444444444", address("Moscow", "Armyanskaya", "1", "1")).
subscriber("Ivanov", "555555555", address("Moscow", "Malaya Bronitsa", "4", "3")).

car("Yugay", "Nissan", "Red", "1000000").
car("Yugay", "Volvo", "Blue", "1200000").
car("Novikov", "Nissan", "Green", "1100000").

deposit("Kerimov", "Sberbank", "0000-0000", "1000000").
deposit("Kerimov", "VTB", "0001-0000", "300000").
deposit("Yugay", "Sberbank", "0000-0001", "500000").

find_car_by_phone(Phone, Surname, Model, Cost) :-
    subscriber(Surname, Phone, _),
    car(Surname, Model, _, Cost).

find_model_by_phone(Phone, Model) :- find_car_by_phone(Phone, _, Model, _).

find_bank_street_by_surname_city(Surname, City, Bank, Street) :-
    subscriber(Surname, _, address(City, Street, _, _)),
    deposit(Surname, Bank, _, _).

goal
find_car_by_phone("222222222", Surname, Model, Cost).
%find_model_by_phone("222222222", Model).
%find_bank_street_by_surname_city("Kerimov", "Lyubertsy", Bank, Street).

```

```

goal
    find_car_by_phone("222222222", Surname, Model, Cost).

[Inactive C:\VIP52\BIN\WIN\32\Obj\goal$000.exe]

Surname=Yugay, Model=Nissan, Cost=1000000
Surname=Yugay, Model=Volvo, Cost=1200000
2 Solutions

```

Рисунок 1 – Фамилия, модель и стоимость машин владельца номера 2222222222

```

goal
    find_model_by_phone("222222222", Model).

[Inactive C:\VIP52\BIN\WIN\32\Obj\goal$000.exe]

Model=Nissan
Model=Volvo
2 Solutions

```

Рисунок 2 – Модель машин владельца номера 2222222222

```

goal
    find_bank_street_by_surname_city("Kerimov", "Lyubertsy", Bank, Street).

[Inactive C:\VIP52\BIN\WIN\32\Obj\goal$000.exe]

Bank=Sberbank, Street=Yubileynaya
Bank=VTB, Street=Yubileynaya
2 Solutions

```

Рисунок 3 – Банки и улица Керимова из Люберец

| № шага | Сравниваемые термы; результат; подстановка, если есть | Дальнейшие действия: прямой ход или откат |
|--------|--|---|
| 1 | Сравнение find_car_by_phone("222222222", Surname, Model, Cost) и subscriber("Kerimov", "111111111", address("Lyubertsy", "Yubileynaya", "17", "2")). Функторы не равны | Прямой ход к следующему предложению |
| ... | ... | ... |
| 12 | Сравнение find_car_by_phone("222222222", Surname, Model, Cost) и find_car_by_phone(Phone, Surname, Model, Cost) :- subscriber(Surname, Phone, _), car(Surname, Model, _, Cost); Phone конкретизируется "222222222". Сцепление Surname к Surname, Model к Model, Cost к Cost. | Прямой ход, к унификации subscriber(Surname, "222222222", _). |
| 13 | Сравнение subscriber(Surname, "222222222", _) и subscriber("Kerimov", "111111111", address("Lyubertsy", "Yubileynaya", "17", "2")). Функторы равны, термы "222222222" и "111111111" не равны | Прямой ход, К следующему предложению. |
| 14 | Сравнение subscriber(Surname, "222222222", _) и subscriber("Yugay", "222222222", address("Lyubertsy", "Yubileynaya", "17b", "18")). Сравниваются термы "222222222" и "222222222". Surname конкретизируется "Yugay" | Прямой ход, К унификации car("Yugay", Model, _, Cost). Surname = "Yugay" |
| ... | ... | ... |
| 20 | Сравнение car("Yugay", Model, _, Cost) и car("Yugay", "Nissan", "Red", "1000000"). Model конкретизируется "Nissan", Cost конкретизируется "1000000". | Откат, переход к следующему предложению. 1 ответ найден Surname = "Yugay" Model = "Nissan" Cost = "1000000" Реконкретизация Model и Cost |
| 21 | Сравнение car("Yugay", Model, _, Cost) и car("Yugay", "Volvo", "Blue", "1200000"). Model конкретизируется "Model", Cost конкретизируется "1200000". | Откат, переход к следующему предложению. 2 ответ найден Surname = "Yugay" Model = "Volvo" Cost = "1200000" Реконкретизация Model и Cost |
| 22 | Сравнение car("Yugay", Model, _, Cost) и car("Novikov", "Nissan", "Green", "1100000"). Функторы равны, термы "Yugay" и "Novikov" не равны | Прямой ход, к следующему предложению Surname = "Yugay" |
| ... | ... | ... |
| 28 | Сравнение car("Yugay", Model, _, Cost) и find_bank_street_by_surname_city(Surname, City, Bank, Street) :- subscriber(Surname,_, address(City, Street, _, _)), deposit(Surname, Bank, _, _). Функторы не равны | Откат, к следующему предложению (относительно шага 14) Surname = "Yugay" Реконкретизация Surname |
| 29 | Сравнение subscriber(Surname, "222222222", _) и subscriber("Ivanov", "333333333", address("Moscow", "Maroseyka", "12", "5")). | Прямой ход, к следующему предложению |

| | | |
|-----|---|--|
| | Функторы равны, термы “2222222222” и “3333333333” не равны | |
| 30 | Сравнение subscriber(Surname, “2222222222”, _) и subscriber("Novikov", "4444444444", address("Moscow", "Armyanskaya", "1", "1")). Функторы равны, термы “2222222222” и “4444444444” не равны. | Прямой ход, переход к следующему предложению. |
| 31 | Сравнение subscriber(Surname, “2222222222”, _) и subscriber("Ivanov", "5555555555", address("Moscow", "Malaya Bronitsa", "4", "3")). Функторы равны, термы “2222222222” и “5555555555” не равны. | Прямой ход, к следующему предложению |
| ... | ... | ... |
| 40 | Сравнение subscriber(Surname, “2222222222”, _) и find_bank_street_by_surname_city(Surname, City, Bank, Street) :- subscriber(Surname,_, address(City, Street, _, _)), deposit(Surname, Bank, _, _). Функторы не равны | Откат, переход к следующему предложению (относительно шага 12) |
| 41 | Сравнение find_car_by_phone(“2222222222”, Surname, Model, Cost) и find_model_by_phone(Phone, Model) :- find_car_by_phone(Phone, _, Model, _). Функторы не равны | Прямой ход, к следующему предложению |
| 42 | Сравнение find_car_by_phone(“2222222222”, Surname, Model, Cost) и find_bank_street_by_surname_city(Surname, City, Bank, Street) :- subscriber(Surname,_, address(City, Street, _, _)), deposit(Surname, Bank, _, _). Функторы не равны | Откат, вывод результатов (см рисунок 1) |

Теоретическая часть

Что такое терм?

Основной элемент языка. Терм — это

1. константа (число, символьный атом, строка),
2. переменная (именованная, анонимная),
3. составной терм $f(t_1, \dots, t_m)$, где f — функтор, t_1, t_2, \dots, t_m — термы.

Что такое предикат в матлогике (математике)?

Предикат — это высказывание, содержащее одно или несколько переменных.

Что описывает предикат в Prolog?

Предикат — это отношение, определяемое процедурой. Процедурой в Prolog называется совокупность правил, заголовки которых имеют одно и то же имя и одну и ту же арность. Процедура описывает совокупность правил отношения.

Назовите виды предложений в программе и приведите примеры таких предложений из Вашей программы. Какие предложения являются основными, а какие — не основными? Каковы: синтаксис и семантика (формальный смысл) этих предложений (основных и неосновных)?

Предложения: факты и правила.

Факт: `deposit("Yugay", "Sberbank", "0000-0001", "500000").`

Правило: `find_model_by_phone(Phone, Model) :- find_car_by_phone(Phone, _, Model, _).`

Предложение является основным, если не содержит переменных, иначе — не основным.

Предложения имеют вид $A : - B_1, \dots, B_n$.

A — заголовок, B_1, \dots, B_n — тело. Фактом называется правило с пустым телом.

Каковы назначение, виды и особенности использования переменных в программе на Prolog? Какое предложение БЗ сформулировано в более общей — абстрактной форме: содержащее или не содержащее переменных?

Назначение переменных — передача информации «во времени и в пространстве».

Виды переменных: именованные и анонимные. Именованная переменная уникальна в пределах предложения, анонимная — любая уникальна. В процессе выполнения программы именованные переменные могут конкретизироваться (связываться со значением). Неименованные переменные не связываются со значениями. Переменные в ходе выполнения программы могут реконкретизироваться.

Переменные в момент фиксации утверждений в программе, обозначая некоторый неизвестный объект из некоторого множества объектов, не имеют значения. Значения для переменных могут быть установлены Prolog-системой только в процессе поиска ответа на вопрос, т.е. реализации программы.

Предложение БЗ, содержащее переменные, является более общим, так как переменные не имеют значения и могут конкретизироваться в ходе выполнения программы.

Что такое подстановка?

Подстановка — это замена одного терма на другой.

Что такое пример терма? Как и когда строится? Как Вы думаете, система строит и хранит примеры?

Пусть $\Theta = \{x_1 = t_1, x_2 = t_2, \dots, x_n = t_n\}$ — подстановка, тогда результат применения подстановки к терму обозначается: $A\Theta$. Применение подстановки заключается в замене каждого вхождения переменной x_i на соответствующий терм. Терм B называется примером терма A , если существует такая подстановка Θ , что $B = A\Theta$.